拉丁超立方抽样的自适应高斯小孔成像蝴蝶优化算法 *

徐 杰[a]，鲁海燕[a, b†]，赵金金[a]，侯新宇[a]，卢梦蝶[a]

(江南大学 a. 理学院; b. 无锡市生物计算工程技术研究中心，江苏 无锡 214122)

摘 要：针对蝴蝶优化算法存在种群多样性差、寻优精度低、收敛速度慢的不足，提出拉丁超立方抽样的自适应高斯小孔成像蝴蝶优化算法。首先利用拉丁超立方抽样种群初始化策略以提高种群的多样性，从而增强算法的全局搜索能力；然后引入在不同进化时期自动调节搜索范围的自适应最优引导策略，平衡算法的全局和局部搜索能力，从而提升算法的寻优精度；最后采用高斯小孔成像策略，对最优个体进行扰动，使得种群个体向最优个体靠近，以进一步提升算法的寻优精度并加快算法的收敛速度。通过对 14 个基准测试函数进行仿真实验以及 Wilcoxon 秩和检验，结果表明改进算法的寻优精度、收敛速度、稳定性和可扩展性等性能均得到了较大提高。

关键词：蝴蝶优化算法；拉丁超立方抽样；自适应惯性权重；高斯小孔成像；高维优化

中图分类号：TP301 doi: 10.19734/j.issn.1001-3695.2022.02.0054

Self-adaptive Gaussian keyhole imaging butterfly optimization algorithm based on Latin hypercube sampling

Xu Jie[a], Lu Haiyan[a, b†], Zhao Jinjin[a], Hou Xinyu[a], Lu Mengdie[a]

(*a. School of Science*, *b. Wuxi Engineering Technology Research Center for Biological Computing*, *Jiangnan University*, *Wuxi Jiangsu* 214122, *China*)

Abstract: Aiming at the shortcomings of Butterfly optimization algorithm, such as poor population diversity, low optimization accuracy and slow convergence speed, this paper proposed a self-adaptive Gaussian keyhole imaging butterfly optimization algorithm based on Latin hypercube sampling. Firstly, it used a Latin hypercube sampling population initialization strategy to enhance the population diversity and thereby improve the overall search ability of the algorithm. Then, it introduced the self-adaptive optimal guidance strategy, which can dynamically adjust the search range in different evolutionary periods to balance the global and local search capabilities and hence improve the optimization accuracy of the algorithm. Finally, it adopted a Gaussian keyhole imaging strategy to disturb the optimal individuals, making the individuals of the population moving close to the optimal individuals, so as to further improve the solution accuracy and speed up the convergence of the algorithm. Through simulation experiments and Wilcoxon rank sum tests using 14 benchmark functions, the results showed that the performance of the improved algorithm is greatly enhanced in terms of optimization accuracy, convergence speed, stability and scalability.

Key words: butterfly optimization algorithm; Latin hypercube sampling; self-adaptive inertia weight; Gaussian keyhole imaging; high dimensional optimization

0 引言

通过模拟自然现象的物理规律及自然界各类生物种群的生活习性和行为特点而构造的一类随机优化算法，被称之为智能优化算法。智能优化算法通常分为三类：群体智能算法，进化算法和基于自然现象的算法[1]。群体智能算法如粒子群算法[2]，海鸥优化算法[3]等；进化算法如遗传进化算法[4]，差分进化算法[5]等；闪电连接过程算法[6]，模拟退火算法[7]等则是基于自然现象的优化算法。这些算法为解决大量存在于计算科学、工程科学、管理科学等领域的复杂问题提供了新的求解途径。

蝴蝶优化算法(butterfly optimization algorithm，BOA)是 2019 年由 Arora 等人[8]提出的一种新的智能优化算法。由于该算法因具有原理简单、参数较少、在求解函数最优值方面具有较好性能等优点，因此被广泛研究并用于粒子滤波器优化、特征选择、传感器节点部署、支持向量机参数优化、故障诊断等实际应用问题的研究中[9~13]。但是和其他元启发式优化算法一样，蝴蝶优化算法自身也存在着一些不足，如算法的种群多样性差、寻优精度不高、收敛速度慢等，这在一定程度上限制了蝴蝶优化算法的理论发展和应用范围。为此，研究者提出了不同的改进策略。文献[14]采用信息交叉共享机制策略使算法跳出局部最优，提高算法的收敛速度。文献[15]提出新型模糊决策策略并且构建“虚拟蝴蝶”来提升算法的搜索能力。文献[16]通过在花蜜位置引入疯狂因子以增加种群多样性，获取更好的最优解，从而提高了算法的寻优精度。文献[17]采用动态调整转换概率策略以平衡全局和局部搜索，提高了算法的搜索效率。文献[18]通过引入正余弦算法作为局部算子来提升种群多样性，从而避免算法陷入局部最优。文献[19]将分段权重引入蝴蝶算法中，以此来平衡全局勘探和局部开发，提升了算法的寻优精度。文献[20]通过构建蝴蝶之间的反馈共享交流网络，提升了算法跳出局部最优的能力。文献[21]通过引入混合惯性权重策略来提高种群地多样性。文献[22]采用自适应余切惯性权重和逐维变异策略来增强算

收稿日期：2022-02-26；修回日期：2022-04-11 基金项目：国家自然科学基金资助项目(61772013，61402201)；江苏省青年基金资助项目(BK20190578)

作者简介：徐杰(1993-)女，河南周口人，硕士研究生，主要研究方向为最优化与控制；鲁海燕(1970-)，女(通信作者)，山东淄博人，副教授，博士，主要研究方向为组合最优化、智能算法(luhaiyan@jiangnan.edu.cn)；赵金金(1997-)，女，河南偃师人，硕士研究生，主要研究方向为最优化与控制；侯新宇(1998-)，男，山东济南人，硕士研究生，主要研究方向为最优化与控制；卢梦蝶(1999-)，女，湖北安陆人，硕士研究生，主要研究方向为最优化与控制.

法的自适应性和种群多样性，从而提升算法的寻优精度和收敛速度。文献[23]通过在算法搜索阶段引入正弦和余弦因子来增强算法的全局搜索能力，从而提升了算法的寻优精度。上述方法尽管对 BOA 的寻优能力有所提升，但算法大多都需要大量的迭代次数，同时对蝴蝶优化算法在高维函数上性能的相关研究较少，而且从未将概率统计的相关知识应用到蝴蝶优化算法性能的研究中。

针对以上问题，本文提出了拉丁超立方抽样的自适应高斯小孔成像蝴蝶优化算法。首先利用拉丁超立方抽样种群初始化策略以提高种群的多样性，从而增强算法的全局搜索能力；然后，为平衡算法的全局和局部搜索能力，提升算法的寻优精度，引入在不同进化时期自动调节搜索范围的自适应最优引导策略；最后采用高斯小孔成像策略，以进一步提升算法的寻优精度并加快算法的收敛速度。实验表明本文所提出的算法在迭代次数较少的情况下不仅提高了算法的寻优精度而且进一步提高了算法的收敛速度。

1　蝴蝶优化算法

通常情况下，自然界中蝴蝶通过嗅觉感知其他蝴蝶散发的香味强弱来调整自身的飞行方向，向着食物源移动，从而实现蝴蝶间的信息交互以此来达到蝴蝶个体自身位置的更新。由于真实的蝴蝶觅食行为有很多不确定的因素，无法在算法中具体体现出来，因此 Arora 等人[8]在两个理想化的条件下建立了蝴蝶优化算法：第一，假设每只蝴蝶都能产生不同程度的香味并且能嗅到来自不同方向的香味，使得蝴蝶个体之间能够相互吸引。这种香味的强度与蝴蝶个体的距离、自然的温度、风速等外界刺激强度的影响有关，其中刺激强度由适应度值决定。用于在蝴蝶迭代过程中分享自身的相关信息。第二，假设每只蝴蝶都有一定的概率嗅到来自最优蝴蝶个体的香味并朝向最优蝴蝶位置方向移动，否则若蝴蝶嗅不到来自最优蝴蝶个体的香味，蝴蝶个体将随机移动。蝴蝶个体向最优个体移动的阶段称为全局搜索阶段，随机移动的阶段则称为局部搜索阶段。全局搜索和局部搜索相互转换的概率称之为转换概率 P，取值为 0.8。蝴蝶产生的芳香度的具体表示如下：

$$f_i^t = c_i^t \times I_i^a \tag{1}$$

$$c_i^{t+1} = c_i^t + \frac{0.025}{c_i^t \times \text{Max_iter}} \tag{2}$$

式(1)中，f_i^t 是第 i 个蝴蝶在 t 代感知到的芳香度，c_i^t 是第 i 个蝴蝶在 t 代的感知因子，I 是刺激的强度，a 是与感知形态有关的幂指数，表示香味的吸收变化。其中刺激强度 I 是由需要优化的函数的适应度值决定的，a 与 c 的取值范围为 $[0,1]$。

此外当 $r<P$ 时，蝴蝶处于全局搜索阶段即蝴蝶个体是向适应度最好的蝴蝶也即全局最优蝴蝶所在位置的方向移动，具体位置移动表达式如下：

$$x_i^{t+1} = x_i^t + (r^2 \times g^* - x_i^t) \times f_i^t \tag{3}$$

其中，x_i^t 是第 i 个蝴蝶在第 t 代的位置；g^* 是种群中最优蝴蝶的位置；f_i^t 是第 i 个蝴蝶在第 t 代感知到的芳香度；r 是 $[0,1]$ 之间的随机数。

当 $r \ge P$ 时，蝴蝶处于局部搜索阶段即蝴蝶个体的移动方向是随机的，具体位置移动表达式如下：

$$x_i^{t+1} = x_i^t + (r^2 \times x_j^t - x_k^t) \times f_i^t \tag{4}$$

其中，x_i^t 是第 i 个蝴蝶在第 t 代的位置；x_j^t，x_k^t 是种群中第 j 个蝴蝶，第 k 个蝴蝶的位置；f_i^t 是第 i 个蝴蝶在 t 代感知到的芳香度；r 是 $[0,1]$ 之间的随机数。

2　拉丁超立方抽样的自适应高斯小孔成像蝴蝶优化算法

本文对蝴蝶算法进行改进，提出了拉丁超立方抽样的自适应高斯小孔成像蝴蝶优化算法(self-adaptive Gaussian keyhole imaging butterfly optimization algorithm based on Latin hypercube sampling, SGLBOA)，该算法首先引入拉丁超立方抽样初始化策略，从而提升算法的全局搜索能力；其次，引入在不同进化时期自动调节搜索范围的自适应最优引导策略以提高算法的寻优精度；最后利用高斯小孔成像策略对最优个体进行扰动，从而进一步提升算法的寻优精度并加快算法的收敛速度。

2.1　拉丁超立方抽样初始化策略

拉丁超立方抽样(Latin hypercube sampling, LHS)是 Mckay 等人[24]于 1979 年提出的一种从多元参数分布中近似随机抽样的方法。在抽样数不多的情况下，随机抽样不能很好地将样本分散到整个区间。与随机抽样不同，拉丁超立方抽样具有均匀分层的特性以及可以在较少抽样的情况下，得到尾部的样本值等优点[25]。

算法的收敛速度和精度在一定程度上是受初始种群好坏的影响的，随机产生的初始种群无法保证搜索空间分布的合理性以及种群的多样性，而拉丁超立方抽样的均匀分层和等概率抽样的特点可以保证其产生的变量覆盖整个分布空间，所以将拉丁超立方抽样应用于算法的初始化可以使初始种群个体尽可能地覆盖整个搜索空间，以提高初始种群的多样性，从而提升算法的寻优性能[26~28]。

由于蝴蝶优化算法的初始种群是随机产生的，因此也难以保证搜索空间分布的合理性以及种群的多样性。为解决这一问题，本文提出的 SGLBOA 算法通过引入拉丁超立方抽样，将搜索空间均匀划分，使初始种群能够较好地覆盖整个搜索空间，从而提高种群的多样性，提升算法的整体搜索能力。SGLBOA 算法种群初始化具体步骤如下：

a) 确定种群规模 N 和维数 D。

b) 设定变量 X 区间为 $[\text{lb},\text{ub}]$，其中 lb 和 ub 分别是变量的上界和下界。

c) 将变量 X 的区间 $[\text{lb},\text{ub}]$ 划分为 N 个相等的子区间。

d) 在每一维里各个子区间中随机抽取一个点。

e) 将抽取的每一维的点组合形成初始种群。

为了直观地展示拉丁超立方抽样初始化策略的有效性，本文通过实验分别采用随机初始化方式和拉丁超立方抽样初始化方式对蝴蝶种群进行初始化，得到如图 1 所示的种群分布图。

由图 1(a)可以看出，采用原 BOA 算法中随机方式初始化种群时，种群分布较为集中，边缘区域蝴蝶个体分布较少，此时种群多样性较差；而使用拉丁超立方抽样初始化得到的种群在整个搜索空间的分布则较为均匀。这从直观上说明了拉丁超立方抽样初始化策略的有效性，本文实验部分也对拉丁超立方抽样初始化策略的有效性进行了验证。

2.2　自适应最优引导策略

为了更好地平衡鸡群算法的全局和局部搜索能力，文献[29]引入了非线性递减的自适应惯性权重因子，其表达式如下：

$$a(t) = t \times \frac{\ln w_{\max} - \ln w_{\min}}{t_{\max}} - \ln w_{\max} \tag{5}$$

$$w(t) = e^{-(a(t))} \tag{6}$$

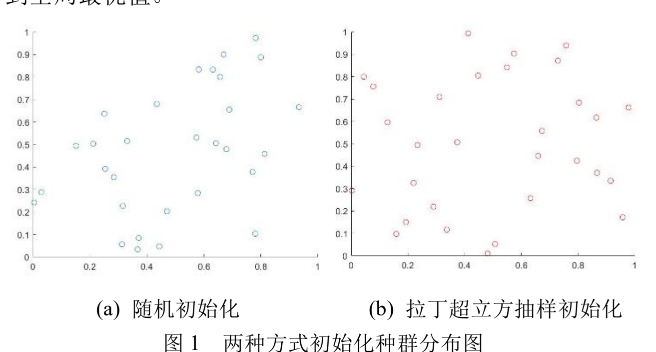
其中，$t_{\max}$ 为总迭代次数；$w(t)$ 为第次 t 迭代的惯性权重因子；$w_{\max}$ 和 $w_{\min}$ 为惯性权重因子的最大值和最小值($w_{\max}=0.9$，$w_{\min}=0.4$)。

受文献[29]的启发，本文将上述自适应惯性权重因子进行改进后引入蝴蝶优化算法中以平衡算法的全局和局部搜索能力，改进后的自适应惯性权重因子为

$$w_1(t) = e^{-(a_1(t))} \tag{7}$$

$$a_1(t) = t \times \frac{\ln w_{\max}}{\ln w_{\min}} - \ln w_{\max} \tag{8}$$

其中：$w_{\max}=0.9$，$w_{\min}=0.4$。在算法迭代前期，权重因子较大，从而使蝴蝶个体有较大的搜索范围，不仅能扩大搜索区域，还能有利于跳出局部极值；而在算法后期搜索阶段权重因子较小，有利于提高蝴蝶群体的局部搜索能力，从而有利于找到全局最优值。



(a) 随机初始化　　　(b) 拉丁超立方抽样初始化

图 1　两种方式初始化种群分布图

Fig. 1　Distribution of initialized population in two ways

蝴蝶优化算法在全局搜索式(4)中引入最优蝴蝶个体 g^*，引导蝴蝶向最优个体方向移动。受此启发可知，要提高蝴蝶优化算法的收敛速度和收敛精度，需要充分发挥最优蝴蝶个体 g^* 的作用，引导种群个体尽可能地向最优个体的方向移动。同时为了避免蝴蝶个体陷入局部最优，引入自适应惯性权重 $w(t)$ 加以平衡。从而改进后的蝴蝶优化算法的全局搜索公式为

$$x_i^{t+1}=w(t)\times g^*+(r^2\times g^*-x_i^t)\times f_i^t \tag{9}$$

对于蝴蝶优化算法局部搜索阶段，为避免算法因较大的随机性而导致搜索效率较低和收敛速度较慢，因此在局部搜索阶段一方面引入最优蝴蝶对蝴蝶个体进行引导，另一方面采用定向变异策略，同时引入判定系数 η，经多次实验可知 η 取 0.9 时，收敛速度和收敛精度最好，故本文中取 $\eta=0.9$。如果 $r>\eta$，则在局部搜索公式中引入扰动因子 θ 以避免因当前最优值的引入而导致算法在局部搜索阶段陷入局部极值；如果 $r\le\eta$，则在局部搜索公式中引入自适应惯性权重因子 $w(t)$ 和黄金比例系数 0.618，使蝴蝶个体自适应的朝着最优个体移动，加快收敛速度。局部搜索阶段的具体表达式为

当 $r>\eta$ 时：

$$x_i^t=\theta\times g^*+(r^2\times x_j^t-x_k^t)\times f_i^t \tag{10}$$

$$\theta=1+\mathrm{trnd}(1)\times\tan\left(\pi\times\left(r-\frac{1}{2}\right)\right) \tag{11}$$

当 $r\le\eta$ 时：

$$x_i^{t+1}=w(t)\times g^*+0.618\times(r^2\times x_j^t-x_k^t)\times f_i^t \tag{12}$$

其中，trnd(1) 是柯西随机数，有助于增加扰动因子 θ 的灵活多样性；r 是 [0,1] 之间的随机数。

2.3　高斯小孔成像策略

小孔成像是一种常见的物理光现象，光源通过小孔从板的一侧倾斜到另一侧，在屏幕上形成倒映像，通过调整小孔屏与物体之间的距离，接收屏上像的大小也会随之发生化[30]。

为了使算法寻找到更广的反向位置，将小孔成像原理简化到反向学习策略中。假设某一空间中，有一个高度为 h 的火焰 p 在 X 轴上的投影为全局最优位置 g^*，坐标轴的上下限为 a，b (当前解空间的上下限)。在基点 O 上放置一个有小孔的小孔屏，火焰通过小孔可以在接收屏上得到一个高度为 h' 的倒映像 p'，此时在 X 轴得到通过小孔成像产生最优位置 g^* 的反向点 g'^*，如图 2 所示。根据相似三角形原理可得：

$$\frac{(a+b)/2-g^*}{g'^*-(a+b)/2}=\frac{h}{h'} \tag{13}$$

令 $\frac{h}{h'}=n$，则 g^* 的反向解为

$$g'^*=\frac{(a+b)}{2}+\frac{(a+b)}{2n}-\frac{g^*}{n} \tag{14}$$

将惯性权重 $w(t)$ 以及均值为 0，方差为 1 的高斯随机数 normrnd(0,1)引入到 g'^* 中，得到 g^* 的最终反向解为

$$g''^*=w(t)\times\frac{(a+b)}{2}+\frac{(a+b)}{2n}-\frac{\mathrm{normrnd}(0,1)\times g^*}{n} \tag{15}$$

其中，n 为参数，经多次实验可知 n 取 2000 时，收敛速度和收敛精度最好，故本文中取 $n=2000$。

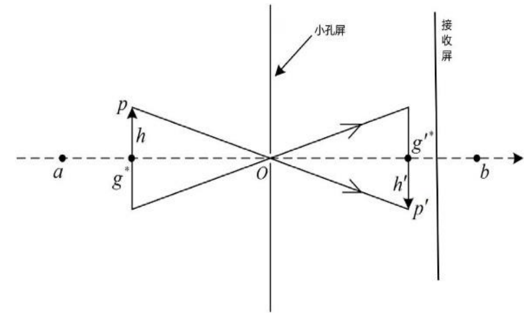


图 2　小孔成像反向学习示意图

Fig. 2　Schematic diagram of keyhole imaging reverse learning

对于蝴蝶优化算法，在搜索后期蝴蝶个体越来越多地聚集在最优蝴蝶的周围，因此 BOA 倾向于早熟收敛。而引入小孔成像反向学习策略可以增加种群的多样性，从而提升蝴蝶优化算法的全局搜索能力。

2.4　SGLBOA 算法流程及时间复杂度分析

2.4.1 SGLBOA 算法流程

SGLBOA 算法流程如图 3 所示。

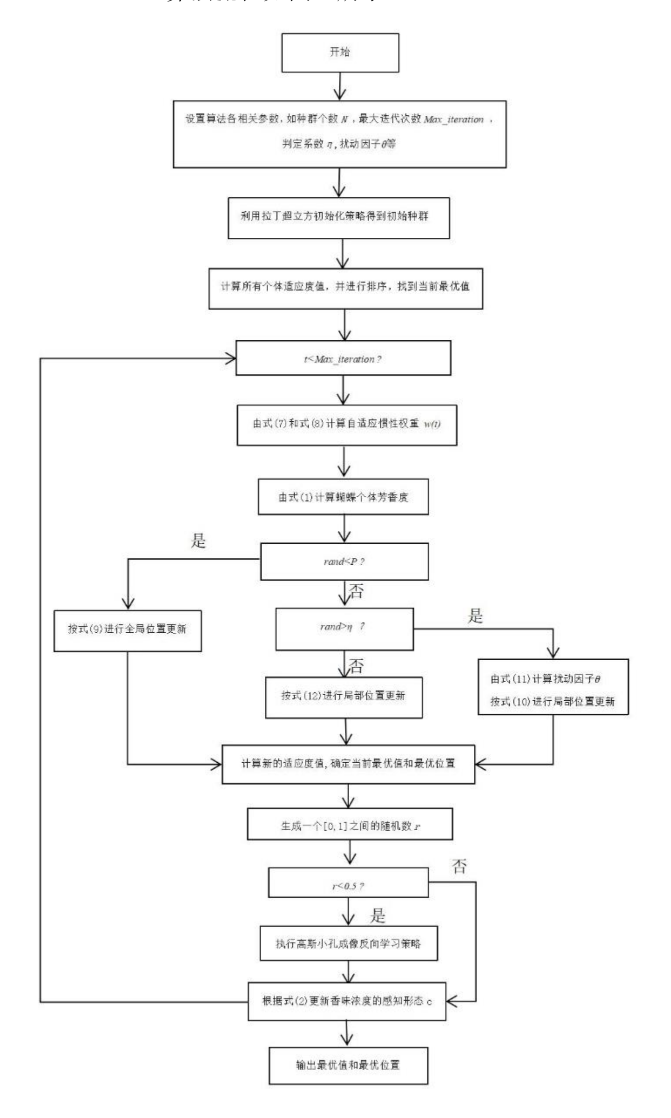


图 3　SGLBOA 算法流程图

Fig. 3　Flow chart of SGLBOA

2.4.2 SGLBOA 算法的时间复杂度分析

由文献[31]可知蝴蝶优化算法在种群大小为 N 、维数为 D 时的时间复杂度为

$$T = O(D + f(D))$$

在 SGLBOA 算法中，种群大小 N 、维数 D 、各参数设置的时间 t_1 、求解适应度值的时间 $f(D)$ 、新，旧蝴蝶个体适应度值比较替换的时间 t_6 、计算香味浓度的时间复杂度 T_2 、记录最优解的时间复杂度 T_5 和更新感觉形态阶段的时间复杂度 T_6 都与文献[31]相同。拉丁超立方抽样初始化阶段的时间复杂度为

$$T_0 = O(N \times D)$$

因此迭代前期的总体的时间复杂度为

$$\begin{aligned} T_1' &= T_0 + O(t_1 + Nf(D)) + \mathrm{Max_iter} \times (T_2 + T_5 + T_6) \\ &= O(D + f(D)) \end{aligned}$$

在全局搜索阶段，SGLBOA 算法在蝴蝶个体的全局位置更新公式中引入了自适应惯性权重 $w(t)$，假设：有 $m(0 \le m \le N)$ 只蝴蝶进入全局搜索，由式(7)和(8)产生 $w(t)$ 的时间为 k_1，根据式(9)进行位置的逐维更新的时间为 k_2，则全局搜索阶段的时间复杂度为

$$T_2' = O(k_1 + NDk_2 + Nf(D) + t_6) = O(D + f(D))$$

在局部搜索阶段，假设满足 $r > \eta$ 的蝴蝶有 q 只 $(0 \le q \le N - m)$，则满足 $r \le \eta$ 的蝴蝶个数为 $N - m - q$ 。假设由式(11)产生扰动因子 θ 的时间为 k_4，按照式(10)进行位置更新的时间为 k_5， 则此时的时间复杂度为

$$T_3' = O(k_4 + NDk_5 + Nf(D) + t_6) = O(D + f(D))$$

当满足 $r \le \eta$ 时，假设由式(12)对蝴蝶个体进行逐维位置更新的时间为 k_6， 则此时的时间复杂度为

$$T_4' = O(NDk_6) = O(D)$$

此外，假设由式(14)和(15)产生 g^* 的小孔成像反向解 g''^* 的时间为 k_7， 则

$$T_5' = O(k_7)$$

因此可得 SGLBOA 算法的时间复杂度为

$$\begin{aligned} T' &= T_1' + \mathrm{Max_iter} \times \left(T_2' + T_3' + T_4' + T_5' + T_6'\right) \\ &= O(D + f(D)) \end{aligned}$$

基于以上分析可知，本文提出的 SGLBOA 算法和 BOA 算法相比，时间复杂度并未增加。

3　实验仿真及结果分析

实验环境为 windows10，算法基于 Matlab2020b。为了测试 SGLBOA 算法的鲁棒性和有效性，引用 14 个基准测试函数进行测试分析，函数具体特征如表 1 所示。

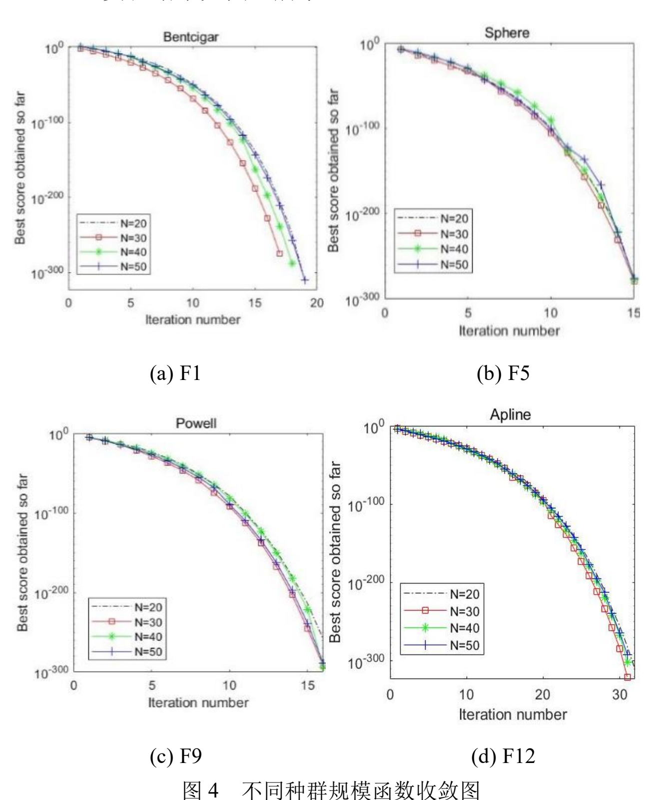
表 1　测试函数

Tab. 1　Test functions

函数	搜索空间	特征	理论最优值
F1：Bentcigar	[-10,10]	单峰	0
F2：Sum of Different Power	[-100,100]	单峰	0
F3：zakharov	[-5,10]	单峰	0
F4：High ConditioneElliptic	[-100,100]	单峰	0
F5：Sphere	[-100,100]	单峰	0
F6：Matyas	[-10,10]	单峰	0
F7：Griewank	[-600,600]	多峰	0
F8：ackley	[-32,32]	多峰	0
F9：Powell	[-4,5]	多峰	0
F10：Rastrigin	[-5.12,5.12]	多峰	0
F11：SchaffersF7	[-100,100]	多峰	0
F12：Apline	[-10,10]	多峰	0
F13：Kowalik	[-5,5]	4 维	3.08E-04
F14：Drop-Wave	[-5.12,5.12]	2 维	-1

3.1　种群规模 N 的选取

在 SGLBOA 算法中，种群规模 N 的不同取值将影响算法的寻优精度和收敛速度。为了在算法中设置合适的 N 值，选取单峰函数 F1 和 F5，多峰函数 F9 和 F12 对 N 值进行敏感性实验。实验中分别取 $N=20$ 、 $N=30$ 、 $N=40$ 、 $N=50$ 和 $D=30$ 。实验结果如图 4 所示。



(a) F1　　(b) F5

(c) F9　　(d) F12

图 4　不同种群规模函数收敛图

Fig. 4　Convergence diagrams with different population sizes

由图 4 可知，无论是对于单峰函数 F1 和 F5 还是对于多峰函数 F9 和 F12，对于相同的迭代次数， $N=30$ 时 SGLBOA 算法寻优精度最好；对于相同的寻优精度，$N=30$ 时 SGLBOA 算法所需的迭代次数最少。综上所述，当种群规模 $N=30$ 时，SGLBOA 算法能达到较好的寻优精度和收敛速度，因此本文后续实验中取 $N=30$ 。

3.2　三种改进策略的有效性分析

为了验证拉丁超立方抽样初始化策略，自适应最优引导策略以及高斯小孔成像反向学习策略的有效性和可行性，本文利用上述 12 个函数对 LBOA，LQBOA 以及 SGLBOA 进行测试。这里 LBOA 是在 BOA 中引入拉丁超立方抽样初始化策略，LQBOA 是在 LBOA 中引入自适应最优引导策略，SGLBOA 是在 LQBOA 中引入高斯小孔成像反向学习策略。各策略对应的算法参数设置均与 BOA 相同，即转换率 $P=0.8$ ，幂指数 $a=0.1$ ，初始感知形态 $c_0=0.01$ 。在 $N=30$ 、 Max_iter=100 、 $D=30$ 的条件下，各算法均独立运行 30 次。实验结果如表 2 所示。

由表 2 可知，LBOA 在均值和标准差方面整体上优于 BOA，这说明加入拉丁超立方抽样初始化策略提升了算法的收敛精度和稳定性，而 LQBOA 在均值和标准差方面比 LBOA 更好，整体达到了最优，这表明加入自适应最优引导策略进一步提升了算法的收敛精度和稳定性。对于函数 F2、F4~F5、F7、F9~F10、F12 和 F14，SGLBOA 的求解结果均为理论最优值 0，同时标准差也为 0，另外在时间方面，SGLBOA 也优于 LBOA 和 LQBOA，这表明高斯小孔成像策略进一步提升算法的寻优精度以及算法的稳定性并加快算法的收敛速度。

3.3　与其他群体智能算法对比分析

为了更全面地验证 SGLBOA 的有效性，文中选取了最近

两年新出现的新型群体智能算法麻雀算法(SSA)[32]和算术优化算法(AOA)[33]，在函数寻优方面性能较好的鲸鱼算法(WOA)[34]和灰狼算法(GWO)[35]以及经典群体智能算法即粒子群算法(PSO)[2]作为对比算法。对比算法参数设置与其对应文献一致。在 $N=30$ 、Max_iter=100 、$D=30$ 的条件下，各算法均独立运行 30 次。实验结果如表 3 所示。

由表 3 可知，对上述测试函数，除函数 F8、F13 和 F14 外，SGLBOA 均能收敛到函数对应的理论最优值。对于函数 F8，SGLBOA 也收敛到了此函数常见的最优值 8.88E-16。此外，对于函数 F1~F6 和 F8~F10，SGLBOA 收敛到理论最优值所用时间相比于其他对比算法而言都较短，并且除函数 F8 外，均值和标准差也都求解到理论最优值 0。由此可知 SGLBOA 在寻优精度、寻优速度方面以及算法稳定性方面均有较好的效果。

表 2　不同改进策略下的实验结果

Tab. 2　Experimental results of different improving strategies

函数	BOA			LBOA			LQBOA			SGLBOA		
	均值	标准差	时间	均值	标准差	时间	均值	标准差	时间	均值	标准差	时间
F1	3.32E-05	4.42E-06	**0.344**	2.52E-05	2.52E-06	0.385	**0.00E+00**	**0.00E+00**	0.473	**0.00E+00**	**0.00E+00**	0.394
F2	3.17E+50	9.38E+50	0.855	5.36E-08	3.85E-08	0.8831	**0.00E+00**	**0.00E+00**	0.5487	**0.00E+00**	**0.00E+00**	**0.4616**
F3	7.41E-06	2.32E-06	**0.3313**	7.36E-06	1.27E-06	0.3561	**0.00E+00**	**0.00E+00**	0.4178	**0.00E+00**	**0.00E+00**	0.3644
F4	2.60E-05	3.82E-06	1.0280	1.98E-05	2.87E-06	1.0725	**0.00E+00**	**0.00E+00**	1.1698	**0.00E+00**	**0.00E+00**	**0.3342**
F5	1.68E-05	1.80E-06	**0.3159**	5.13E-06	4.62E-07	0.3418	**0.00E+00**	**0.00E+00**	0.4117	**0.00E+00**	**0.00E+00**	0.3343
F6	1.47E-06	7.94E-07	**0.2756**	2.58E-07	1.67E-07	0.3161	**0.00E+00**	**0.00E+00**	0.4172	**0.00E+00**	**0.00E+00**	0.3311
F7	3.69E-04	1.80E-03	**0.5548**	3.42E-11	5.13E-11	0.6203	**0.00E+00**	**0.00E+00**	0.6737	**0.00E+00**	**0.00E+00**	0.5910
F8	1.04E-02	6.30E-03	0.4669	2.92E-04	2.47E-05	0.4440	**8.88E-16**	**8.88E-16**	0.4609	**8.88E-16**	**8.88E-16**	**0.3906**
F9	1.37E-05	2.02E-06	0.5732	8.74E-06	1.26E-06	0.6067	**0.00E+00**	**0.00E+00**	0.5151	**0.00E+00**	**0.00E+00**	**0.3920**
F10	1.15E-04	1.15E-04	0.4579	1.06E-05	2.67E-06	0.4353	**0.00E+00**	**0.00E+00**	0.4524	**0.00E+00**	**0.00E+00**	**0.3502**
F11	5.17E-02	9.10E-03	2.5816	1.54E-02	3.40E-03	2.6321	**0.00E+00**	**0.00E+00**	1.7806	**0.00E+00**	**0.00E+00**	**1.7041**
F12	1.30E-03	2.03E-04	0.3899	2.35E-04	2.07E-05	**0.3679**	3.71E-176	0.00E+00	0.4553	**0.00E+00**	**0.00E+00**	0.3809
F13	1.20E-03	1.20E-03	**0.3200**	3.96E-04	6.34E-05	0.3452	**3.50E-03**	**3.00E-03**	0.4546	6.30E-03	4.20E-03	0.3802
F14	-1.65E+05	5.60E+05	**0.5091**	-1.02E+00	1.02E-01	0.5713	**-1.00E+00**	**5.19E-06**	0.6521	**-1.00E+00**	**5.19E-06**	0.5145

表 3　与其他群体智能算法的对比实验结果

Tab. 3　Experimental results and comparison with other swarm intelligence algorithms

函数	算法	SSA	AOA	WOA	GWO	PSO	SGLBOA	函数	算法	SSA	AOA	WOA	GWO	PSO	SGLBOA
F1	最优值	**0.00E+00**	4.23E-36	6.73E-10	3.31E+03	2.48E+07	**0.00E+00**	F8	最优值	**8.88E-16**	**8.88E-16**	4.75E-12	20.79E+00	20.72E+00	**8.88E-16**
	均值	2.42E-64	5.78E-09	1.89E-05	1.39E+04	1.24E+08	**0.00E+00**		均值	**8.88E-16**	9.36E-04	5.10E-03	20.98E+00	20.99E+00	**8.88E-16**
	标准差	1.33E-63	1.77E-08	5.78E-05	8.52E+03	4.78E+07	**0.00E+00**		标准差	**0.00E+00**	3.50E-03	2.79E-02	9.28E-02	1.30E-01	**0.00E+00**
	时间/s	4.4506	0.4493	1.5276	1.3247	0.7644	**0.3941**		时间/s	1.3022	0.4068	0.4429	0.6021	0.6496	**0.3906**
F2	最优值	**0.00E+00**	4.87E-202	1.68E-24	6.40E-04	7.19E+17	**0.00E+00**	F9	最优值	**0.00E+00**	7.18E-36	1.22E-15	31.29E+00	3.25E+04	**0.00E+00**
	均值	2.32E-44	5.02E-16	2.55E-02	3.43E+00	3.46E+23	**0.00E+00**		均值	3.28E-52	2.30E-03	1.88E-08	6.06E+04	2.06E+05	**0.00E+00**
	标准差	1.21E-43	2.72E-15	9.28E-02	8.75E+00	1.76E+24	**0.00E+00**		标准差	1.90E-51	1.19E-02	9.62E-08	6.23E+04	1.44E+05	**0.00E+00**
	时间/s	1.6254	0.6161	0.7166	0.7319	1.3179	**0.4616**		时间/s	1.3450	0.5121	0.5321	0.6288	0.9692	**0.3919**
F3	最优值	**0.00E+00**	2.41E-10	7.59E+04	1.31E+04	2.00E+04	**0.00E+00**	F10	最优值	**0.00E+00**	**0.00E+00**	1.14E-13	28.62E+00	338.88E+00	**0.00E+00**
	均值	1.70E-38	450.71E+00	9.82E+04	3.53E+04	3.32E+04	**0.00E+00**		均值	**0.00E+00**	5.20E-02	2.21E-07	55.96E+00	540.44E+00	**0.00E+00**
	标准差	9.29E-38	2.47E+03	1.25E+04	1.04E+04	8.71E+03	**0.00E+00**		标准差	**0.00E+00**	2.85E-02	9.89E-07	18.70E+00	133.09E+00	**0.00E+00**
	时间/s	1.2530	0.3500	0.4119	0.4182	0.5587	**0.3644**		时间/s	1.3079	0.4184	0.4484	0.6006	0.8167	**0.3502**
F4	最优值	**0.00E+00**	3.56E-27	2.49E-11	5.90E+00	7.55E+05	**0.00E+00**	F11	最优值	**0.00E+00**	1.21E-12	1.79E-06	4.33E-01	3.86E+00	**0.00E+00**
	均值	4.45E-45	7.79E-04	6.69E-06	27.05E+00	3.43E+06	**0.00E+00**		均值	6.65E-12	4.06E-06	5.96E-05	6.00E-01	5.20E+00	**0.00E+00**
	标准差	2.44E-44	2.90E-03	2.63E-05	15.81E+00	1.60E+06	**0.00E+00**		标准差	3.64E-11	1.48E-05	9.87E-05	1.27E-01	5.61E-01	**0.00E+00**
	时间/s	1.6806	0.7465	0.8140	0.8764	1.4982	**1.0860**		时间/s	2.6035	1.5506	1.7544	**1.6437**	2.4154	1.7041
F5	最优值	**0.00E+00**	1.70E-50	3.38E-15	2.90E-03	54.46E+00	**0.00E+00**	F12	最优值	**0.00E+00**	9.32E-25	5.97E-11	6.14E-01	70.31E+00	**0.00E+00**
	均值	1.27E-56	1.70E-04	1.51E-11	1.52E-02	144.17E+00	**0.00E+00**		均值	1.02E-24	2.14E-06	3.02E-07	5.74E+00	144.28E+00	**0.00E+00**
	标准差	6.97E-56	7.41E-04	5.12E-11	1.05E-02	48.75E+00	**0.00E+00**		标准差	5.59E-24	1.15E-05	9.27E-07	6.07E+00	41.17E+00	**0.00E+00**
	时间/s	1.2396	0.3464	0.3548	0.4984	0.7040	**0.3343**		时间/s	1.2424	**0.3682**	0.4162	0.5489	0.5994	0.3809
F6	最优值	**0.00E+00**	**0.00E+00**	3.66E-62	5.36E-28	2.76E-11	**0.00E+00**	F13	最优值	**3.09E-04**	4.79E-04	3.10E-04	3.87E-04	3.91E-04	3.53E-04
	均值	3.60E-61	**0.00E+00**	1.33E-39	3.84E-18	6.19E-09	**0.00E+00**		均值	3.25E-04	2.44E-02	**1.30E-03**	4.7E-03	8.20E-04	6.30E-03
	标准差	1.97E-60	**0.00E+00**	6.72E-39	1.76E-17	2.15E-08	**0.00E+00**		标准差	**1.17E-05**	3.39E-02	2.90E-02	8.00E-03	3.12E-04	4.20E-03
	时间/s	1.2610	0.3404	0.3560	0.4897	0.4967	**0.3311**		时间/s	0.5188	**0.2805**	0.3443	0.2398	0.5147	0.3802
F7	最优值	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	3.22E-14	**0.00E+00**	F14	最优值	**-1.00E+00**	**-1.00E+00**	**-1.00E+00**	**-1.00E+00**	**1.00E+00**	**-1.00E+00**
	均值	**0.00E+00**	**0.00E+00**	**0.00E+00**	3.29E-04	6.31E-10	**0.00E+00**		均值	**-1.00E+00**	**-1.00E+00**	-9.70E-01	-9.69E-01	-9.93E-01	**-1.00E+00**
	标准差	**0.00E+00**	**0.00E+00**	**0.00E+00**	1.80E-03	9.66E-10	**0.00E+00**		标准差	**0.00E+00**	**0.00E+00**	3.24E-02	3.19E-02	1.95E-02	5.19E-06
	时间/s	1.4244	0.5261	**0.5798**	0.6189	0.8608	0.5910		时间/s	0.4247	0.1957	0.2982	**0.1746**	0.4395	0.5145

3.4　改进蝴蝶优化算法在高维测试函数上的性能分析

为了验证 SGLBOA 在高维问题上的有效性，将 SGLBOA 与 BOA 的其他改进算法作对比。为保证算法对比的时效性，本文选取新近出现的在求解函数最优值方面性能较好的蝴蝶改进算法作为对比算法，即 CFSBOA[20]、MSBOA1、MSBOA2 和 SIBOA[23]。其中 MSBOA1 和 MSBOA2 在对应的文献[21]和文献[22]中都被记为 MSBOA，本文为了便于区分将其分别记为 MSBOA1 和 MSBOA2。在 $N=30$，$\mathrm{Max_iter}=100$，算法中的其余参数与对应参考文献中设置一致的条件下。将 6 种算法分别在 $D=1000$、$D=3000$ 和 $D=5000$ 时独立运行 30 次，得到的最优值、平均值、标准差、平均耗时的实验数据如表 4~6 所示。

由表 4 可知，在 $D=1000$ 时，对于函数 F1~F7，F9~F12，SGLBOA 的求解结果均为理论最优值 0，这表明 SGLBOA 在收敛精度上远远高于其他对比算法。而对于函数 F8，虽然没有收敛到理论最优值 0，但也收敛到了常见的最优值 8.88E-16。此外，对于大多数函数，相比于其他改进算法，SGLBOA 寻优时间最短。因此 SGLBOA 在收敛速度上具有一定的优势。

为了直观地验证算法的收敛性能，图 5 给出了 6 种算法在 $D=1000$ 时部分测试函数的收敛图。由该图可知，对于单峰函数 F1、F4~F6 以及多峰函数 F9 和 F10，SGLBOA 在 20 代左右已达到收敛，即使对于较为复杂的多峰函数 F11 和 F12，SGLBOA 在 40 代左右也已收敛，然而对于这些函数，其他对比算法在 100 代时也远未达到收敛；另外，对于相同的迭代次数，其他对比算法的寻优精度也都低于 SGLBOA。这表明 SGLBOA 在收敛速度和寻优精度方面都明显优于其余 5 种对比算法。

由表 5 可知，在 $D=3000$ 时，对于函数 F1、F2、F4~F7、F9~F12，SGLBOA 的求解结果均为理论最优值 0，而对于函数 F8 而言，同样是收敛到了常见的最优值 8.88E-16。同时，对于函数 F2、F8、F9~F11，SGLBOA 是 5 种算法中寻优时间最短的，这表明 SGLBOA 收敛速度较快。对于函数 F4~F6 以及 F12，SGLBOA 虽然寻优时间上不敌其他 4 种算法，但均值和方差是 5 种算法中最小的，这表明 SGLBOA 具有较好的稳定性。

由表 6 可知，在 $D=5000$ 时，对于函数 F1、F2、F4~F7、F9~F12，SGLBOA 的求解结果均为理论最优值 0，而对于函数 F8，依旧是收敛到了常见的最优值 8.88E-16。此外，对于函数 F2，F8~F11，虽然 5 种算法的求解结果均为理论最优值 0，但 SGLBOA 寻优时间最短。而对于函数 F1、F4、F6、F12，SGLBOA 的寻优时间虽然不是 5 种算法中最短的，但 SGLBOA 的寻优精度和稳定性却是 5 种算法中最好的。

此外，由表 4~6 可知，对于大多数函数，改进算法 CFSBOA[20]、MSBOA1、MSBOA2、SIBOA[23]以及 SGLBOA 相比于 BOA，平均耗时都有所增加，但 SGLBOA 的平均耗时增加量最小，而且 SGLBOA 在寻优精度、收敛速度、稳定性以及可扩展性方面均有大幅度提升。

图 6 给出了 SGLBOA 在不同维数 D 时部分测试函数的收敛图，通过该图可以直观地看到随着测试函数维度的升高，SGLBOA 依然保持着较好的寻优性能。这表明 SGLBOA 具有良好的可扩展性。

综合以上分析可知，SGLBOA 在高维问题上的求解结果整体优于其他 5 种对比算法，从而表明 SGLBOA 在高维函数优化问题上具有较好的寻优精度、收敛速度、稳定性和可扩展性。

表 4　不同改进蝴蝶优化算法在 D=1000 的对比实验结果

Tab. 4　Experimental results and comparison of different improved butterfly optimization algorithms with　D=1000

函数	算法	BOA	CFSBOA	MSBOA1	MSBOA2	SIBOA	SGLBOA	函数	算法	BOA	CFSBOA	MSBOA1	MSBOA2	SIBOA	SGLBOA
F1	最优值	2.51E-05	5.50E-83	3.93E-93	1.10E-20	4.582E-53	**0.00E+00**	F7	最优值	1.14E-08	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	均值	3.45E-05	3.10E-70	5.14E-17	7.72E-11	7.03E-49	**0.00E+00**		均值	7.01E-04	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	标准差	4.65E-06	1.58E-69	2.82E-16	3.69E-10	2.32E-48	**0.00E+00**		标准差	2.50E-03	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	时间/s	**2.4868**	3.0016	3.4015	39.5388	2.6354	3.6061		时间/s	**1.0566**	1.6029	1.7189	23.0358	1.2591	2.0233
F2	最优值	--	3.75E-96	--	1.64E-60	4.10E-04	**0.00E+00**	F8	最优值	1.80E-03	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**
	均值	--	2.28E-84	--	1.14E-30	--	**0.00E+00**		均值	3.40E-03	**8.88E-16**	**8.88E-16**	9.22E-13	**8.88E-16**	**8.88E-16**
	标准差	--	6.25E-84	--	6.20E-30	--	**0.00E+00**		标准差	7.49E-04	**0.00E+00**	**0.00E+00**	8.46E-13	**0.00E+00**	**0.00E+00**
	时间/s	--	35.5629	--	313.3699	15.5259	**5.1754**		时间/s	10.7737	7.1906	13.1204	66.9593	7.7307	**4.6400**
F3	最优值	2.51E-19	1.05E-73	9.49E-118	2.44E+06	1.71E-58	**0.00E+00**	F9	最优值	2.05E-05	2.28E-84	2.44E-104	4.38E-28	1.60E-61	**0.00E+00**
	均值	2.13E-08	4.11E-28	2.40E-21	3.57E+06	**1.73E-40**	1.19E+17		均值	2.32E-05	6.43E-74	1.30E-27	1.72E-06	1.01E-52	**0.00E+00**
	标准差	4.41E-08	2.25E-27	1.32E-20	1.99E+06	**9.47E-40**	3.61E+17		标准差	2.13E-06	3.47E-73	7.03E-27	9.43E-06	5.48E-52	**0.00E+00**
	时间/s	**1.3711**	1.5301	1.6819	22.5229	1.3746	1.9299		时间/s	17.9143	21.3937	19.1475	303.5922	17.5327	**7.2015**
F4	最优值	2.31E-05	1.05E-83	1.42E-98	3.56E-22	4.04E-56	**0.00E+00**	F10	最优值	5.78E-06	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	均值	3.36E-05	6.29E-70	5.87E-26	3.63E-14	5.81E-50	**0.00E+00**		均值	2.76E-05	**0.00E+00**	**0.00E+00**	5.30E-11	**0.00E+00**	**0.00E+00**
	标准差	4.34E-06	3.44E-69	3.22E-25	1.38E-13	2.59E-49	**0.00E+00**		标准差	1.46E-05	**0.00E+00**	**0.00E+00**	1.03E-10	**0.00E+00**	**0.00E+00**
	时间/s	**8.0532**	11.8962	8.5681	149.2748	8.2131	8.9430		时间/s	5.4525	9.5266	12.5484	107.3682	4.3755	**2.3056**
F5	最优值	1.93E-05	1.01E-88	3.61E-92	1.93E-51	1.00E-63	**0.00E+00**	F11	最优值	2.05E-02	1.23E-22	1.95E-22	3.22E-13	3.01E-16	**0.00E+00**
	均值	2.44E-05	3.66E-78	7.88E-24	3.23E-30	4.78E-58	**0.00E+00**		均值	2.47E-02	7.40E-21	1.87E-07	2.86E-11	3.33E-15	**0.00E+00**
	标准差	2.39E-06	1.57E-77	4.32E-23	1.21E-29	2.33E-57	**0.00E+00**		标准差	1.80E-03	1.13E-20	9.84E-07	3.04E-11	2.97E-15	**0.00E+00**
	时间/s	**1.0250**	1.4102	1.4445	22.2540	1.1633	1.9457		时间/s	22.7010	32.6799	24.0581	399.7773	22.7013	**13.5322**
F6	最优值	3.20E-07	1.92E-98	2.77E-123	3.86E-80	4.02E-72	**0.00E+00**	F12	最优值	1.80E-03	1.89E-43	1.93E-50	5.12E-22	1.00E-30	**0.00E+00**
	均值	1.43E-06	1.72E-85	1.63E-44	3.32E-61	7.89E-64	**0.00E+00**		均值	2.30E-03	4.04E-38	3.83E-11	1.92E-16	1.58E-27	**0.00E+00**
	标准差	5.62E-07	5.94E-85	6.86E-44	1.64E-60	3.83E-63	**0.00E+00**		标准差	2.45E-04	1.13E-37	2.10E-10	5.54E-16	5.11E-27	**0.00E+00**
	时间/s	**0.8434**	1.1294	1.3750	18.2334	0.9880	1.7284		时间/s	**2.1540**	3.0340	3.6104	41.3533	2.3069	2.6784

表 5　不同改进蝴蝶优化算法在 D=3000 的对比实验结果

Tab. 5　Experimental results and comparison of different improved butterfly optimization algorithms with　D=3000

函数	算法	BOA	CFSBOA	MSBOA1	MSBOA2	SIBOA	SGLBOA	函数	算法	BOA	CFSBOA	MSBOA1	MSBOA2	SIBOA	SGLBOA
F1	最优值	2.82E-05	2.64E-80	3.62E-97	9.58E-23	1.129E-55	**0.00E+00**	F7	最优值	6.51E-10	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	均值	3.43E-05	8.78E-72	1.00E-26	6.36E-14	1.75E-50	**0.00E+00**		均值	6.84E-04	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	标准差	3.57E-06	4.23E-71	4.13E-26	3.26E-13	8.16E-50	**0.00E+00**		标准差	2.60E-03	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	时间/s	**2.6383**	3.4132	3.7957	151.3863	4.7307	5.0902		时间/s	**2.5788**	5.8771	7.3489	167.2194	3.7848	8.5168
F2	最优值	--	4.54E-94	--	3.06E-60	1.53E-05	**0.00E+00**	F8	最优值	7.77E-04	8.88E-16	8.88E-16	8.88E-16	8.88E-16	**8.88E-16**
	均值	--	4.98E-82	--	1.22E-30	--	**0.00E+00**		均值	9.58E-04	8.88E-16	8.88E-16	3.4757e-12	8.88E-16	**8.88E-16**
	标准差	--	1.97E-81	--	6.66E-30	--	**0.00E+00**		标准差	8.45E-05	0.00E+00	0.00E+00	2.2317e-12	0.00E+00	**0.00E+00**
	时间/s	--	100.0776	--	3265.1555	71.9326	**27.0344**		时间/s	23.7051	15.9489	32.6699	383.3505	20.8926	**10.9917**
F3	最优值	3.24E-05	**1.42E-62**	3.55E+08	4.24E-08	1.46E-34	4.66E+14	F9	最优值	1.68E-05	4.58E-87	5.49E-93	1.54E-37	4.77E-65	**0.00E+00**
	均值	5.40E+21	**3.97E-29**	6.97E+18	6.98E+04	2.11E-27	5.83E+22		均值	2.10E-05	3.54E-73	1.78E-30	9.96E-16	2.15E-59	**0.00E+00**
	标准差	2.51E+21	**2.17E-28**	2.34E+19	5.13E+04	6.63E-27	1.74E+23		标准差	2.01E-06	1.35E-72	9.74E-30	4.47E-15	8.04E-59	**0.00E+00**
	时间/s	5.717	**5.0131**	7.7128	187.3159	5.7349	7.6199		时间/s	51.1329	54.9721	45.666	2074.5216	42.5611	**25.3653**
F4	最优值	3.04E-05	1.35E-79	6.18E-88	5.18E-22	3.12E-55	**0.00E+00**	F10	最优值	1.09E-06	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	均值	3.57E-05	4.55E-67	9.90E-19	8.37E-12	1.09E-49	**0.00E+00**		均值	6.62E-06	**0.00E+00**	**0.00E+00**	2.4253e-13	**0.00E+00**	**0.00E+00**
	标准差	3.23E-06	2.49E-66	5.42E-18	4.57E-11	3.87E-49	**0.00E+00**		标准差	5.62E-06	**0.00E+00**	**0.00E+00**	1.3284e-12	**0.00E+00**	**0.00E+00**
	时间/s	**38.4395**	60.7089	45.0255	2030.0856	40.2137	41.6311		时间/s	23.6051	22.8525	31.2439	399.7311	16.2036	**8.673**
F5	最优值	2.05E-05	5.36E-90	1.24E-89	9.86E-45	1.21E-63	**0.00E+00**	F11	最优值	2.03E-02	8.23E-24	4.52E-19	6.11E-14	3.73E-16	**0.00E+00**
	均值	2.58E-05	3.84E-74	5.01E-27	5.06E-28	2.26E-57	**0.00E+00**		均值	2.47E-02	7.39E-21	2.93E-09	5.31E-11	3.21E-15	**0.00E+00**
	标准差	2.66E-06	2.09E-73	2.75E-26	2.35E-27	9.33E-57	**0.00E+00**		标准差	2.06E-03	1.15E-20	1.32E-08	9.47E-11	4.63E-15	**0.00E+00**
	时间/s	**2.7723**	5.1965	5.8508	192.7266	3.1643	7.8005		时间/s	72.4257	102.9056	74.4172	3543.2641	64.372657	**40.5479**
F6	最优值	9.48E-09	8.56E-99	1.08E-108	4.36E-77	8.99E-73	**0.00E+00**	F12	最优值	1.93E-03	8.84E-46	9.38E-43	1.65E-22	7.57E-31	**0.00E+00**
	均值	5.34E-07	1.68E-83	9.83E-48	1.55E-64	7.58E-68	**0.00E+00**		均值	2.28E-03	2.15E-38	2.44E-14	4.31E-17	1.57E-28	**0.00E+00**
	标准差	2.46E-07	9.17E-83	3.75E-47	8.02E-64	2.21E-67	**0.00E+00**		标准差	1.64E-04	5.31E-38	1.34E-13	1.05E-16	4.51E-28	**0.00E+00**
	时间/s	**3.1369**	4.1799	5.4877	142.1456	3.9304	7.0109		时间/s	**5.4693**	8.0107	9.4143	244.6403	5.7093	7.7365

表 6　不同改进蝴蝶优化算法在 D=5000 的对比实验结果

Tab. 6　Experimental results and comparison of different improved butterfly optimization algorithms with　D=5000

函数	算法	BOA	CFSBOA	MSBOA1	MSBOA2	SIBOA	SGLBOA	函数	算法	BOA	CFSBOA	MSBOA1	MSBOA2	SIBOA	SGLBOA
F1	最优值	2.77E-05	2.15E-81	6.12E-90	8.47E-24	1.11E-54	**0.00E+00**	F7	最优值	2.85E-11	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	均值	3.43E-05	3.37E-68	9.38E-21	6.10E-13	1.94E-51	**0.00E+00**		均值	3.21E-08	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	标准差	3.31E-06	1.84E-67	5.14E-20	3.28E-12	5.71E-51	**0.00E+00**		标准差	4.09E-08	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	时间/s	7.7651	8.4849	8.8395	391.4922	**7.3067**	10.1121		时间/s	**5.9967**	9.2575	10.2146	324.8185	6.8590	10.2753
F2	最优值	--	5.29E-93	--	8.03E-58	6.42E-05	**0.00E+00**	F8	最优值	5.53E-04	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**
	均值	--	8.17E-80	--	1.96E-25	--	**0.00E+00**		均值	6.46E-04	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**
	标准差	--	4.36E-79	--	1.07E-24	--	**0.00E+00**		标准差	5.59E-05	**0.00E+00**	**0.00E+00**	3.87E-12	**0.00E+00**	**0.00E+00**
	时间/s	--	99.4272	--	8296.3997	83.2912	**27.489**		时间/s	32.9034	24.0397	44.565	718.5489	25.1074	**14.9316**
F3	最优值	2.59E-05	**1.20E-61**	6.81E+08	3.81E+04	4.16E-31	3.86E+18	F9	最优值	1.60E-05	3.03E-87	4.41E-96	8.69E-30	7.95E-64	**0.00E+00**
	均值	1,94E+22	**3.04E-26**	1.45E+19	1.37E+05	7.58E-25	1.28E+25		均值	2.38E-05	4.47E-74	9.18E-40	2.74E-14	3.53E-57	**0.00E+00**
	标准差	3.78E+22	**1.16E-25**	4.51E+19	7.81E+04	1.69E-24	1.99E+25		标准差	2.68E-06	2.34E-73	4.34E-39	1.25E-13	1.90E-56	**0.00E+00**
	时间/s	9.6822	**9.1985**	11.8947	400.2919	9.2538	9.4833		时间/s	64.5646	75.5385	63.5889	4959.0585	60.8643	**24.8883**
F4	最优值	2.87E-05	2.12E-82	2.09E-83	6.55E-21	6.39E-54	**0.00E+00**	F10	最优值	3.72E-06	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	均值	3.60E-05	1.45E-69	2.43E-17	5.65E-11	3.44E-49	**0.00E+00**		均值	1.93E-05	**0.00E+00**	**0.00E+00**	2.47E-11	**0.00E+00**	**0.00E+00**
	标准差	4.22E-06	7.91E-69	1.33E-16	3.09E-10	1.26E-48	**0.00E+00**		标准差	7.47E-06	**0.00E+00**	**0.00E+00**	8.46E-11	**0.00E+00**	**0.00E+00**
	时间/s	**38.8434**	55.9658	40.7198	3044.8077	38.9418	42.0427		时间/s	34.093	33.3223	42.9037	1312.8956	25.3992	**13.2231**
F5	最优值	1.91E-05	1.63E-88	2.64E-84	4.90E-58	1.88E-62	**0.00E+00**	F11	最优值	1.33E-02	7.47E-23	1.71E-19	6.61E-14	1.93E-16	**0.00E+00**
	均值	2.69E-05	4.85E-76	7.29E-16	1.85E-28	2.09E-56	**0.00E+00**		均值	1.46E-02	9.04E-21	3.04E-08	1.18E-11	2.79E-15	**0.00E+00**
	标准差	3.21E-06	1.74E-75	3.99E-15	8.33E-28	9.86E-56	**0.00E+00**		标准差	5.93E-04	1.32E-20	1.64E-07	1.67E-11	3.02E-15	**0.00E+00**
	时间/s	**7.3230**	9.0121	9.2273	412.2919	7.7815	10.4181		时间/s	102.4434	145.7607	105.7969	9219.1753	188.959827	**57.0347**
F6	最优值	1.86E-08	7.17E-97	2.21E-106	1.31E-90	2.82E-73	**0.00E+00**	F12	最优值	2.09E-03	2.49E-45	1.50E-40	2.40E-23	5.57E-31	**0.00E+00**
	均值	7.05E-07	7.09E-84	2.90E-36	6.36E-65	1.80E-66	**0.00E+00**		均值	2.55E-03	3.66E-38	3.35E-11	7.49E-17	5.98E-28	**0.00E+00**
	标准差	3.24E-07	3.86E-83	1.49E-35	2.15E-64	9.32E-66	**0.00E+00**		标准差	2.10E-04	1.39E-37	1.78E-10	1.85E-16	2.12E-27	**0.00E+00**
	时间/s	**6.1523**	7.3095	8.8976	280.3555	6.3400	9.2357		时间/s	**13.6142**	18.798	23.1732	884.6742	14.8423	17.9933

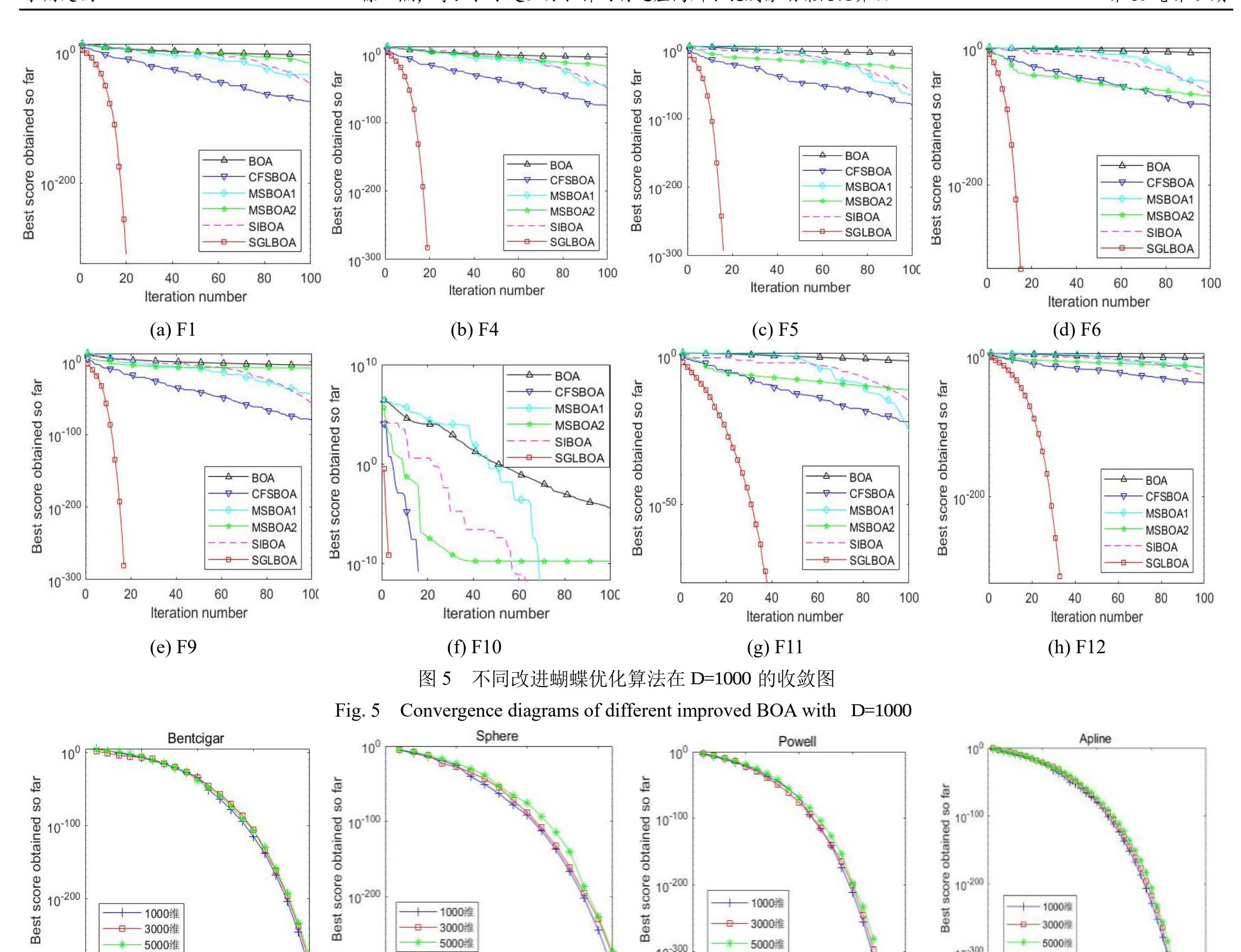


图 5 不同改进蝴蝶优化算法在 D=1000 的收敛图

Fig. 5 Convergence diagrams of different improved BOA with D=1000

图 6 SGLBOA 算法在不同维数下部分测试函数的收敛图

Fig. 6 Convergence diagrams of SGLBOA on some test functions in different dimensions

3.5 Wilcoxon 秩和检验

对于改进算法，通常需要利用统计检验来验证，相对于现有算法而言，改进算法是否具有显著性。为判断 SGLBOA 的寻优结果是否在统计上与其他算法的最佳结果显著不同，Wilcoxon 秩和检验在 5%的显著性水平下进行也即当 p 值小于 5%时，“R”为“+”，即拒绝 H0 假设，说明两种算法的差异性显著；否则就是接受 H0 假设，说明两种算法在整体上是相同的。表 7 给出了在 14 个基准测试函数下，SGLBOA 对 BOA[8]、CFSBOA[20]、MSBOA1、MSBOA2 以及 SIBOA[23]的秩和检验 p 值。从表 7 中可以观察到大多数的 p 值是小于 5%，因此总体上说明 SGLBOA 与其他 5 种算法之间差异性显著，SGLBOA 明显优于其他 5 种对比算法。

表 7 Wilcoxon 秩和检验结果

Tab. 7 Wilcoxon rank sum test results

函数	BOA		CFSBOA		MSBOA1		MSBOA2		SIBOA	
	p	R	p	R	p	R	p	R	p	R
F1	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F2	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F3	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F4	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F5	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F6	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F7	1.2118e-12	+	NA	=	NA	=	NA	=	NA	=

函数	BOA		CFSBOA		MSBOA1		MSBOA2		SIBOA	
	p	R	p	R	p	R	p	R	p	R
F8	1.2118e-12	+	NA	=	0.3337	-	8.6979e-07	+	NA	=
F9	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F10	1.2118e-12	+	NA	=	NA	=	0.0056	-	NA	=
F11	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F12	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+	1.2118e-12	+
F13	7.0430e-07	+	1.8567e-09	+	1.6351e-05	+	8.9934e-11	+	2.4386e-09	+
F14	4.2770e-12	+	NA	=	NA	=	NA	=	NA	=

注：“R”为显著性判断结果，“+”“-”“=”分别表示 MSBOA 的性能优于、劣于和相当对比算法；“NA”表示不适用，无法进行显著判断

4 结束语

本文在蝴蝶优化算法的基础上提出拉丁超立方抽样的自适应高斯小孔成像蝴蝶优化算法。通过将概率统计中的拉丁超立方抽样策略应用于算法的种群初始化，提高种群的多样性，以提升算法的全局搜索能力；自适应最优引导策略的引入能够自动调节不同进化时期的搜索范围，较好地平衡算法的全局和局部搜索能力，从而提升算法的寻优精度；引入高

斯小孔成像策略，进一步提升算法的寻优精度并加快算法的收敛速度。14 个测试函数的仿真实验结果表明 SGLBOA 算法具有较好的寻优精度、收敛速度、稳定性以及可扩展性。同时由于 SGLBOA 算法在高维优化问题上具有较好的性能，未来可进一步研究将算法应用于大规模优化问题中。

参考文献：

[1] Achikkulath P, Hussain S. Quantum chaotic butterfly optimization algorithm with ranking strategy for constrained optimization problems [J]. IEEE Access, 2021, 9: 114587-114608.

[2] Kenney J, Eberhart R C. Particle swarm optimization [C]// Proc of IEEE International Conference on Neural Networks, 1995: 1942-1948.

[3] Xu Le, Mo Yuanbin, Lu Yanyue, *et al.* Improved seagull optimization algorithm combined with an unequal division method to solve dynamic optimization problems [J]. Processes, 2021, 9: 1037.

[4] Abdullah Y C, Iklim G B, Birdal S. A new approach using the genetic algorithm for parameter estimation in multiple linear regression with long-tailed symmetric distributed error terms: an application to the Covid-19 data [J]. Chemometrics and Intelligent Laboratory Systems, 2021, 216: 104372.

[5] Ning Guiying, Zhou Yongquan. Application of improved differential evolution algorithm in solving equations [J]. International Journal of Computational Intelligence Systems, 2021, 14 (1) .

[6] Nematollahi A F, Rahiminejad A, Vahidi B. A novel physical based meta-heuristic optimization method known as lightning attachment procedure optimization [J]. Applied Soft Computing, 2017, 59: 596-621.

[7] Zhang Jingrui, Li Zhuoyun, Wang Beibei, *et al.* Within-day rolling optimal scheduling problem for active distribution networks by multi-objective evolutionary algorithm based on decomposition integrating with thought of simulated annealing [J]. Energy, 2021, 223: 120027.

[8] Arora S, Singh S. Butterfly optimization algorithm: a novel approach for global optimization [J]. Soft Computing, 2019, 23 (3): 715-73

[9] 刘慧，姜雨汐. 融合分数阶和蝴蝶优化的改进粒子滤波算法 [J/OL]. 小型微型计算机系统: 1-6. (2021-12-01) [2022-2-13]. https://kns. cnki. net/kcms/detail/21. 1106. TP. 20211130. 1302. 006. html (Liu Hui, Jiang Yuxi. Improved particle filter algorithm incorporating fractional-order and butterfly optimization [J/OL]. Journal of Chinese Computer Systems: 1-6. (2021-12-01) [2022-2-13]. https://kns. cnki. net/kcms/detail/21. 1106. TP. 20211130. 1302. 006. html.)

[10] Sharma S, Saha A K. m-MBOA: a novel butterfly optimization algorithm enhanced with mutualism scheme [J]. Soft Computing, 2020, 24 (7): 4809-4827.

[11] 张孟健，汪敏，王霄，等. 混合粒子群-蝴蝶算法的 WSN 节点部署研究 [J/OL]. 计算机工程与科学: 1-9. (2021-9-16) [2022-2-13]. https://kns. cnki. net/kcms/detail/43. 1258. TP. 20210916. 1048. 002. html. (Zhang Mengjian, Wang Min, Wang Xiao, *et al.* Research on node deployment in WSN based on hybrid particle swarm-butterfly algorithm [J/OL]. Computer-Engineering & Science: 1-9. (2021-9-16) [2022-2-13]. https://kns. cnki. net/kcms/detail/43. 1258. TP. 20210916. 1048. 002. html.)

[12] Wen Lei, Cao Yang. A hybrid intelligent predicting model for exploring household CO2 emissions mitigation strategies derived from butterfly optimization algorithm [J]. Science of the Total Environment, 2020, 727: 138572.

[13] 赵玲玲，王群京，陈权，等. 基于 IBBOA 优化 BP 神经网络的变压器故障诊断 [J]. 电工电能新技术，2021, 40 (9): 39-46. (Zhao Lingling, Wang Qunjing, Chen Quan, *et al.* Fault diagnosis of transformer based on BP neural network optimized by IBBOA [J]. Advanced Technology of Electrical Engineering and Energy, 2021, 40 (9): 39-46.)

[14] Jun Luo, Qin Tian, Xu Meng. Reverse guidance butterfly optimization algorithm integrated with information cross-sharing [J]. Journal of Intelligent and Fuzzy Systems, 2021, 41: 3463–3484.

[15] Ali M, Mahsa M. Enhanced butterfly optimization algorithm with a new fuzzy regulator strategy and virtual butterfly concept [J]. Knowledge-Based Systems, 2021, 228: 107-291.

[16] 王依柔，张达敏，徐航，等. 基于自适应扰动的疯狂蝴蝶算法 [J]. 计算机应用研究, 2020, 37 (11): 3276-3280. (Wang Yirou, Zhang Damin, Xu Hang, *et al.* Crazy butterfly algorithm based on adaptive perturbation [J]. Application Research of Computers, 2020, 37 (11): 3276-3280.)

[17] 刘凯，代永强. 融合变异策略的自适应蝴蝶优化算法 [J]. 计算机应用研究, 2021, 39 (1): 1-9. (Liu Kai, Dai Yongqiang, Adaptive butterfly optimization algorithm based on mutation strategies [J]. Application Research of Computers, 2021, 39 (1): 1-9.)

[18] 高文欣，刘升，肖子雅，等. 全局优化的蝴蝶优化算法 [J]. 计算机应用研究, 2020, 37 (10): 2966-2970. (Gao Wenxin, Liu Sheng, Xiao Ziya, *et al.* Butterfly optimization algorithm for global optimization [J]. Application Research of Computers, 2020, 37 (10): 2966-2970.)

[19] 李守玉，何庆，杜逆索. 分段权重和变异反向学习的蝴蝶优化算法 [J]. 计算机工程与应用, 2021, 57 (22): 92-101. (Li Shouyu, He Qing, Du Nisuo, Piecewise weight and mutation opposition-based learning butterfly optimization algorithm [J]. Computer Engineering and Applications, 2021, 57 (22): 92-101.)

[20] 李守玉，何庆，杜逆索. 混沌反馈共享和群体协同效应的蝴蝶优化算法 [J/OL]. 计算机科学与探索: 1-12. (2021-1-28) [2022-2-13]. https://kns. cnki. net/kcms/detail/11. 5602. TP. 20210128. 1109. 014. html. (Li Shouyu, He Qing, Du Nisuo. Butterfly optimization algorithm for chaotic feedback sharing and group synergy [J/OL]. Journal of Frontiers of Computer Science and Technology: 1-12. (2021-1-28) [2022-2-13]. https://kns. cnki. net/kcms/detail/11. 5602. TP. 20210128. 1109. 014. html.)

[21] 陈俊，何庆. 基于余弦相似度的改进蝴蝶优化算法 [J]. 计算机应用, 2021, 41 (9): 2668-2677. (Chen Jun, He Qing. Improved butterfly optimization algorithm based on cosine similarity [J]. Journal of Computer Applications, 2021, 41 (9): 2668-2677.)

[22] 宁杰琼，何庆. 混合策略改进的蝴蝶优化算法 [J]. 计算机应用研究, 2021, 38 (6): 1718-1723+1738. (Ning Jieqiong, He Qing. Mixed strategy to improve butterfly optimization algorithm [J]. Application Research of Computers, 2021, 38 (6): 1718-1723+1738.)

[23] 王依柔，张达敏. 融合正弦余弦和无限折叠迭代混沌映射的蝴蝶优化算法 [J]. 模式识别与人工智能, 2020, 33 (7): 660-669. (Wang Yirou, Zhang Damin. Butterfly optimization algorithm combining sine cosine and iterative chaotic map with infinite collapses [J]. Pattern Recognition and Artificial Intelligence, 2020, 33 (7): 660-669.)

[24] McKay M D, Beckman R J, Conover W J, *et al.* Comparison the three methods for selecting values of input variable in the analysis of output from a computer code [J]. Technometrics, 1979, 21 (2): 239.

[25] 魏锋涛，张洋洋，黎俊宇，等. 基于动态分级策略的改进正余弦算法 [J]. 系统工程与电子技术, 2021, 43 (6): 1596-1605. (Wei Fengtao, Zhang Yangyang, Li Junyu, *et al.* Improved sine cosine algorithm based on dynamic classification strategy [J]. Systems Engineering and Electronics, 2021, 43 (6): 1596-1605.)

[26] He Zhenxue, Pan Yuhua, Wang Kejian, *et al.* Area optimization for MPRM logic circuits based on improved multiple disturbances fireworks algorithm [J]. Applied Mathematics and Computation, 2021, 399: 126008.

[27] Siti J R, Hasliza A R, Khairul N A R, *et al.* A hybrid modified method of

the sine cosine algorithm using Latin hypercube sampling with the cuckoo search algorithm for optimization problems [J]. Electronics, 2020, 9: 1786.

[28] Alaa T, Wolfram S. Population initialization techniques for evolutionary algorithms for single-objective constrained optimization problems: deterministic vs. stochastic techniques [J]. Swarm and Evolutionary Computation, 2021, 67: 100952.

[29] 顾艳春，鲁海燕，向蕾，等. 自适应动态学习鸡群优化算法 [J]. 计算机工程与应用, 2020, 56 (20): 36-45. (Gu Yanchun, Lu Haiyan, Xiang Lei, *et al.* Adaptive dynamic learning chicken swarm optimization algorithm [J]. Computer Engineering and Applications, 2020, 56 (20): 36-45.)

[30] 张达敏，徐航，王依柔，等. 嵌入 Circle 映射和逐维小孔成像反向学习的鲸鱼优化算法 [J]. 控制与决策, 2021, 36 (5): 1173-1180. (Zhang Damin, Xu Hang, Wang Yirou, *et al.* Whale optimization algorithm for embedded Circle mapping and one-dimensional oppositional learning based small hole imaging [J]. Control and Decision, 2021, 36 (5): 1173-1180.)

[31] 刘景森，马义想，李煜. 改进蝴蝶算法求解多维复杂函数优化问题 [J]. 电子学报, 2021, 49 (6): 1068-1076. (Liu Jingsen, Ma Yixiang, Li Yu. Improved butterfly algorithm for multi-dimensional complex function optimization problem [J]. Acta Electronic Sinica, 2021, 49 (6): 1068-1076.)

[32] Xue Jiankai, Shen Bo. A novel swarm intelligence optimization approach: sparrow search algorithm [J]. Systems Science and Control Engineering, 2020, 8 (1): 22-34.

[33] Abualigah L, Diabat A, Mirjalili, *et al.* The arithmetic optimization algorithm [J]. Computer Methods in Applied Mechanics & Engineering, 2021, 376: 113609.

[34] Mirjalili S, Lewis A. The whale optimization algorithm [J]. Advances in Engineering Software, 2016, 95: 51-67.

[35] Mirjalili S, Mirjalili S M, Lewis, *et al.* Grey wolf optimizer [J]. Advances in Engineering Software, 2014, 69: 46-61.